University of Zurich[UZH]

Zurich Open Repository and Archive
University of Zurich
University Library
Strickhofstrasse 39
CH-8057 Zurich
www.zora.uzh.ch

Year: 2015

Secure and efficient wireless sensor networks

Schmitt, Corinna ; Stiller, Burkhard

Abstract: There exists a multitude of implemented, as well as envisioned, use cases for the Internet of Things (IoT) and Wireless Sensor Networks (WSN). Some of these use cases would benefit from the collected data being globally accessible to: (a) authorized users only; and (b) data processing units through the Internet. Much of the data collected, such as location or personal identifiers, are of a highly sensitive nature. Even seemingly innocuous data (e.g., energy consumption) can lead to potential infringements of user privacy.

Posted at the Zurich Open Repository and Archive, University of Zurich
ZORA URL: https://doi.org/10.5167/uzh-116554
Journal Article
Published Version

Originally published at:
Schmitt, Corinna; Stiller, Burkhard (2015). Secure and efficient wireless sensor networks. ERCIM News, 2015(101):18-19.

Number 101 April 2015

ERCIM NEWS

www.ercim.eu

Special theme:

The Internet of Things and The Web of Things

Also in this issue:

Keynote:
by Cees Links, Founder & CEO GreenPeak Technologies

Research and Innovation:
Mesh Joinery:
A Method for Building Fabricable Structures

Secure and Efficient Wireless Sensor Networks

by Corinna Schmitt and Burkhard Stiller

There exists a multitude of implemented, as well as envisioned, use cases for the Internet of Things (IoT) and Wireless Sensor Networks (WSN). Some of these use cases would benefit from the collected data being globally accessible to: (a) authorized users only; and (b) data processing units through the Internet. Much of the data collected, such as location or personal identifiers, are of a highly sensitive nature. Even seemingly innocuous data (e.g., energy consumption) can lead to potential infringements of user privacy.

The infrastructure of the Internet of Things (IoT) with its diversity of devices and applications, as well as the trend towards a separation of sensor network infrastructure and applications exacerbates security risks. A true end-to-end security solution is therefore required to achieve an adequate level of security for IoT. Protecting data once they leave the boundaries of a local network is not sufficient, especially when private and high-risk information are effected.

However, IoT is no longer limited to servers, routers, and computers with manifold resources. It also includes constrained devices – motes –, which are very limited in memory (approximately 10-50 KByte RAM and 100-256 KByte ROM), computational capacity, and power (supported by just a few AA batteries). These limited resources do not reduce the need to support end-to-end security and secure communications, but they make it much harder to meet these requirements. Depending on the specific resources of these devices, the goal of secure WSNs is to support end-to-end security by a two-way authentication, an efficient data transport solution for the data, and a controlled data access, supporting the mobility of today's users. Thus, different components were developed in the construction of SecureWSN and are illustrated in Figure 1. All components support hardware from different vendors with different resources. Different security solutions (TinySAM, TinyDTLS [2], or TinyTO [3]) were developed that are based on known algorithms from IP networks, like DTLS and BCK, and required adaptation (e.g., complexity) to fit these resources whilst supporting a heterogonous network structure.

End-to-end Security

Today, with so much personal information online, end-to-end security is essential in many situations. This represents the challenge for constrained devices usually used in WSNs. SecureWSN tackles this challenge by developing three solutions for different types of resources.

TinyDTLS protects data from its source to the sink supporting confidentiality, integrity, and authenticity. RSA-capable (Rivest-Shamir-Adleman) devices are authenticated via X.509 certificates during the key-exchange in a two-way authentication handshake. Constrained devices perform a variant of the Transport Layer Security (TLS) pre-shared key algorithm. The data sink authenticates via certificate either directly with the mote or with an Access Control Server (ACS). ACS grants tickets to authenticated devices with sufficient rights. Motes request connection from their communication partner where key establishment is based on DTLS [2].

In comparison, TinyTO uses a Bellare-Canetti-Krawcyk (BCK) handshake with pre-shared keys. For the key generation, key exchange, signatures, and

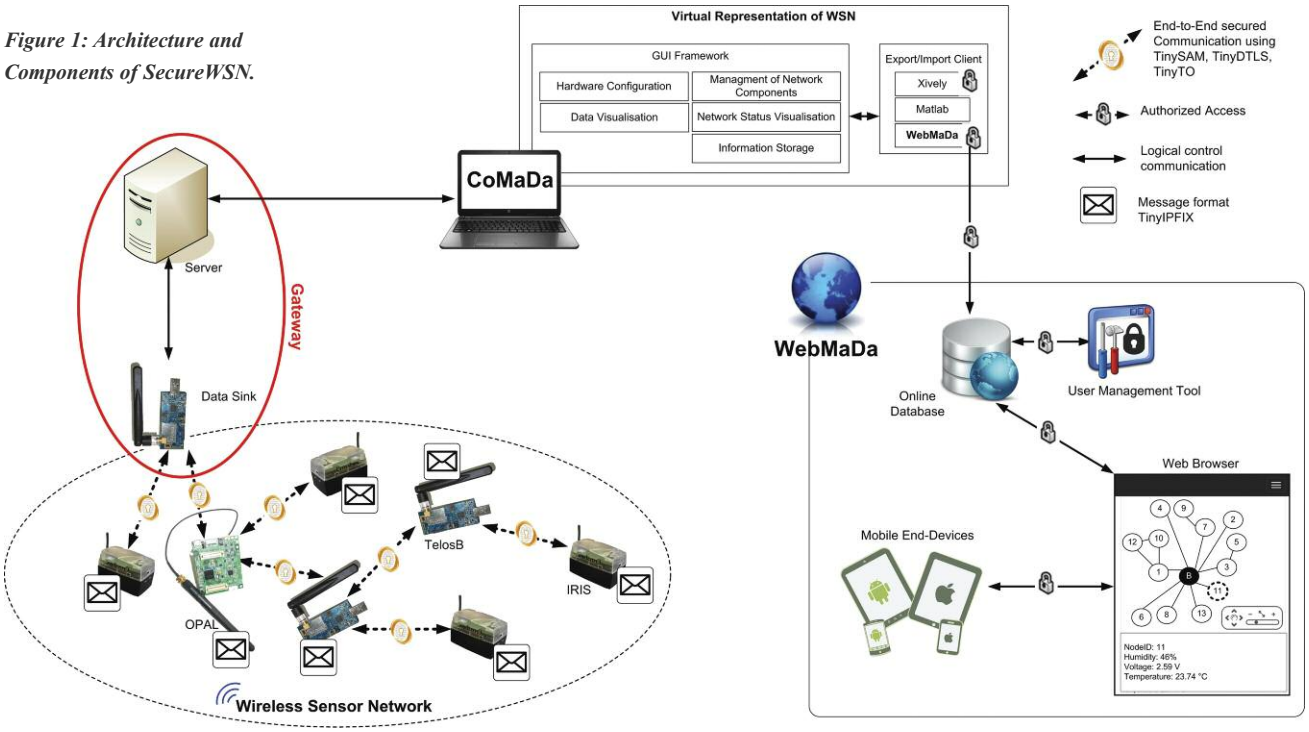

Figure 1: Architecture and Components of SecureWSN.

encryption, the Elliptic Curve Cryptography (ECC) is used [3]. Thus, this solution saves resources and does not require a Certificate Authority (CA). It was shown that 192-bit ECC keys are as secure as 1024-bit to 2048-bit RSA keys, which makes TinyTO a suitable alternative to TinyDTLS, supporting the same security functionality.

As sufficient resources are not always available to support the end-to-end security requirement, TinySAM was developed to support a one-way authentication. TinySAM uses the Otway-Rees key establishment protocol modified by the Abadi and Needham algorithm, where all nodes have an AES (Advanced Encryption Standard) key pair known by the key server. Two nodes build an individual session key pair for secure data exchange.

Owing to the diversity of applications and the amount of collected data, all these solutions also support aggregation in order to use the limited bandwidth (102 byte on MAC layer in IEEE 802.15.4) and energy as efficient as possible.

Data collected in WSNs consists of stable meta-information and sensor readings periodically measured and sent out in one message resulting in redundancy. Thus, the push-based Internet Protocol Flow Information Export (IPFIX) protocol serves optimization by dividing data into two small messages (template record and data record). The resulting TinyIPFIX protocol includes special template records for motes and supports aggregation. Necessary header compression options were developed to reduce the overhead by required IPFIX headers [2].

Additionally, the SecureWSN approach includes the WSN configuration, management, and data handling of the WNS's owner by 'clicking buttons', all termed CoMaDa. CoMaDa works with a virtual representation of the real WSN network, displaying in real-time: (1) data collected and (2) the network status, as well as allowing for (3) mote updates (e.g., the degree of aggregation). The dedicated WebMaDa component [1] publishes the WSN data in the Internet and allows anyone who is authorized and has the appropriate credentials and rights, to view the WSNs.

Conclusions

SecureWSN consists of different modules supporting different end-to-end security modes, efficient data transport, aggregation, and controlled data access functionality. These solutions, which are currently available, are highly flexible, since each mechanism that is implemented can be selected depending on the requirements of the applications and hardware. As such, the approach of SecureWSN benefits any kind of IoT application that demands secure end-to-end support.

Continued work in this area will include further module developments and enhancements, such as pull requests and ECC optimizations. Parts of SecureWSN were developed within EU projects SmartenIT and FLAMINGO and are part of the standardization process.

Links:

http://www.csg.uzh.ch/research/SecureWSN.html

http://tools.ietf.org/html/draft-schmitt-ace-twowayauth-for-iot-01

References:

[1] M. Keller: "Design and Implementation of a Mobile App to Access and Manage Wireless Sensor Networks", Master Thesis, Universität Zürich, Communication Systems Group, Department of Informatics, Zürich, Switzerland, November 2014.

[2] T. Kothmayr, et al.: "DTLS Based Security and Two-way Authentication for the Internet of Things", Ad Hoc Networks, Elsevier, Vol. 11, No. 8, November 2013, pp 2710-2723.

[3] M. Noack: "Optimization of Two-way Authentication Protocol in Internet of Things", Master Thesis, Universität Zürich, Communication Systems Group, Department of Informatics, Zürich, Switzerland, August 2014.

Please contact:

Corinna Schmitt, Burkhard Stiller
Universität Zürich, Switzerland,
Tel: +41 44 635 7585,
+41 44 635 6710
E-mail: schmitt@ifi.uzh.ch,
stiller@ifi.uzh.ch

Creating Internet of Things Applications from Building Blocks

by Frank Alexander Kraemer and Peter Herrmann

Reactive Blocks is a toolkit to create, analyse and implement reactive applications. It helps developers to get concurrency issues right, reuse existing solutions, and brings model checking to the common programmer.

Internet of Things (IoT) applications connect hardware to communicate within a network, often with constrained resources. Therefore, even simple use cases quickly become very complex. In addition, IoT applications often combine several technologies, and only few programmers have the skill set to cover them all.

Model-driven development and model checking provide solutions to these problems. However, these solutions are barely used by programmers. Reasons for this are that model checking requires deep knowledge in formal methods to produce an appropriate input model that can be analysed. In addition, model-driven approaches often fail to cover details in code in a suitable way.

With these concerns in mind, we developed the Reactive Blocks tool for event-driven and concurrent applications. It has its origins at the Department of Telematics at the Norwegian University of Science and Technology (NTNU),